

## **BAB III**

### **PELAKSANAAN KERJA MAGANG**

#### **3.1. Divisi, Kedudukan dan Koordinasi**

##### **3.1.1 Divisi**

Diperlihatkan pada struktur organisasi dari PT Kalyana Adikarya Abadi bahwa saat kerja magang ditempatkan pada divisi *software development*. Divisi tersebut dikoordinir oleh seorang technical lead yang bernama Kenneth Aloysius. Untuk menjalankan proyek nantinya setiap orang akan dibagi kedalam sebuah tim untuk menanganinya. Saat menangani proyek dari PT XYZ dikerjakan bersama dengan Ryan Natanael sebagai bisnis analis dan Pak Dadi Irsan selaku *project manager*.

Proyek yang akan dikerjakan ialah mengenai pembuatan RPA untuk melakukan *testing* pada aplikasi dari PT XYZ. Aplikasi yang akan di *testing* merupakan aplikasi *point of sales* (POS). *Scope* dari *testing* meliputi beberapa fungsi dari aplikasi tersebut. Fungsi yang akan di tes contohnya seperti pengetesan halaman login, *stok opname*, transaksi dengan *mix payment*, dan lainnya.

##### **3.1.2 Kedudukan dan Koordinasi**

Saat praktik kerja magang dilakukan peran yang diberikan adalah sebagai RPA *developer* pada divisi *software development*. Untuk mengerjakan sebuah proyek RPA *developer* bekerjasama dengan bisnis

analisis. Selain itu untuk melaporkan pengembangan proyek juga akan dilakukan laporan mingguan kepada *project manager*.

Sebagai *RPA developer* nantinya akan diberikan tugas untuk membuat *workflow*. *Workflow* akan dibuat berdasarkan *test scenario* yang diberikan oleh tim QA dari PT XYZ. Untuk pembuatan tahapan dari setiap *test scenario* tim dari Kalyana akan bekerjasama dengan PIC dari QA PT XYZ. Setelah itu dibuat *workflow* pada aplikasi UiPath.

### **3.2. Tugas yang Dilakukan**

Selama kurang lebih tiga bulan menjalankan praktik kerja magang pada PT Kalyana Adikarya Abadi beberapa tugas sudah dijalankan. Untuk tugas pertama untuk setiap *RPA developer* yaitu menjalankan *training RPA developer foundation*. Setelah menyelesaikan *training foundation* diwajibkan juga untuk menyelesaikan *training RPA developer advance*. Kedua *training* tersebut merupakan persyaratan bagi *RPA developer* untuk ikut andil dalam sebuah proyek.

Sambil menunggu dimulainya proyek diberikan terlebih dahulu mempelajari mengenai *RPA testing* menggunakan UiPath. Selain itu juga diberikan tugas untuk mempelajari mengenai UiPath *mobile automation* yang merupakan fitur baru dari UiPath yang dapat digunakan untuk automasi *mobile* dengan platform Android maupun iOS. Dikarenakan *client* akan menggunakan sistem *on premise* maka Orchestrator *on premise* dan UiPath Test Manager juga harus dipelajari untuk disiapkan di komputer *client*.

Setelah menyelesaikan sertifikasi dan tugas lainnya nantinya akan diikutsertakan dalam sebuah proyek dengan *client* dari PT Kalyana Adikarya Abadi dengan *role* sebagai RPA *developer*. Proyek yang akan dijalankan yaitu *testing* aplikasi POS milik PT XYZ. Sebelum menjalankan proyek dilakukan meeting POC terlebih dahulu untuk melihat demo dari pihak PT XYZ mengenai aplikasinya. Setelah itu akan dilakukan *setup* aplikasi yang dibutuhkan pada komputer *client*. *Setup* dilakukan secara *remote* pada komputer *server* dan *user client*. Ketika semua kebutuhan untuk *development* telah terpenuhi tim Kalyana akan berkoordinasi dengan tim QA PT XYZ untuk mengetahui *flow* dari beberapa *test scenario* untuk kebutuhan demo. Untuk lebih jelas uraian kerja magang diperlihatkan oleh tabel 3.1 uraian kerja magang.

**Tabel 3. 1 Uraian Kerja Magang**

Minggu	Aktivitas
1-2	Training (UI Path Developer Foundation)
3-4	Training (UI Path Developer Advance)
5	Training (RPA Testing)
6-7	UI Path Mobile Automation
8	UIPath Orchestrator - On Premise
9-10	UIPath Test Manager
1-12	Meeting POC & Pembuatan Requirement Analysis

Penjabaran lebih detail mengenai timeline pada tabel 3.1 uraian kerja magang akan diperlihatkan pada sub bab berikut.

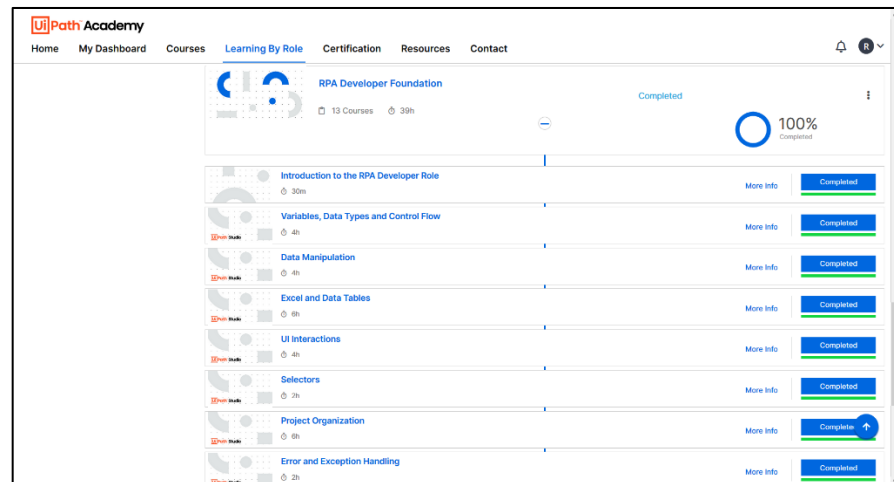
### 3.2.1. Minggu 1 & 2 : Training RPA Developer Foundation

Pada hari pertama dilakukan pengenalan dahulu dengan rekan-rekan kerja satu kantor. Setelah itu pembimbing lapangan yakni Kenneth Aloysius memberikan instruksi mengenai detail pekerjaan yang harus dilakukan. Hal yang pertama dilakukan yaitu mempersiapkan email yang telah dibuatkan oleh kantor. Setelah email siap maka selanjutnya pendaftaran dapat langsung dilakukan pada *website* UiPath. Setelah melakukan pendaftaran pada *website* UiPath maka pengguna akan mendapatkan akses untuk mengunduh *software* UiPath dan menjalankan latihan RPA *developer foundation*.

Sebelum memulai pelatihan RPA *developer foundation* perlu dilakukan instalasi UiPath Studio terlebih dahulu. Saat akan melakukan instalasi ada 3 jenis UiPath Studio yang dapat dipilih yaitu Studio, Studio Pro, dan Studio X. Studio X dipilih untuk instalasi dikarenakan memiliki fitur yang paling lengkap dibandingkan yang lainnya. Setelah instalasi selesai maka pelatihan *developer foundation* dapat langsung dimulai.

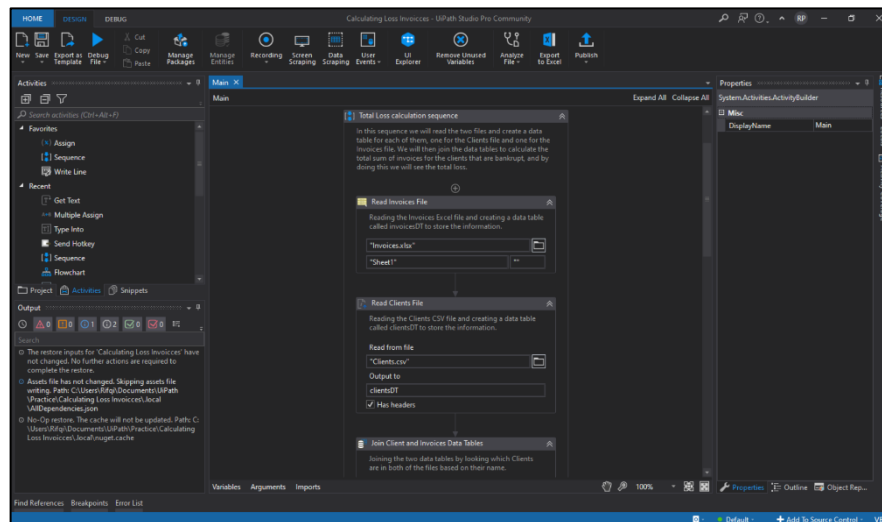
Pelatihan yang disediakan oleh UiPath dibagi kedalam modul-modul yang topiknya dapat terlihat pada dashboard UiPath *academy* seperti yang terlihat pada gambar 3.1 *dashboard* UiPath *academy*. Pada setiap modul mencakup beberapa modul tertentu yang dijelaskan menggunakan tulisan maupun video. Bukan hanya sekedar penjelasan tetapi pengetahuan pengguna juga akan diuji oleh latihan soal yang pada setiap akhir sesi tiap

materi. Pelatihan ini juga mengulas penggunaan Visual Basic .NET yang akan digunakan pada *development* RPA dengan UiPath.



**Gambar 3. 1 Dashboard UiPath Academy**

Materi yang disediakan pada *RPA developer foundation* meliputi *variables, data types, data manipulation, excel automation, pdf automation* dan masih banyak lainnya. Pada setiap topik akan diberikan contoh *workflow* RPA yang dapat diikuti. Jika pengguna merasa bingung dengan tutorial pada video maka disediakan juga *workflow* yang dapat diunduh dan dibuka pada UiPath Studio sehingga tidak mengalami kebuntuan pada suatu topik. Gambar 3.2 tampilan UiPath Studio memperlihatkan tampilan dari UiPath Studio.

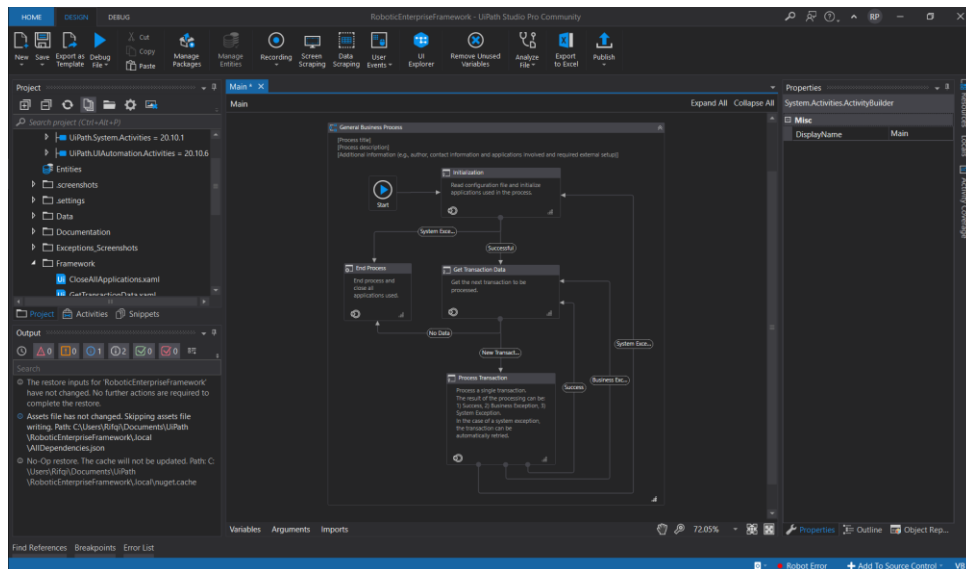


**Gambar 3. 2 Tampilan UiPath Studio**

Setiap peserta harus menyelesaikan setiap materi dan quiz pelatihan *developer foundation* sebagai persyaratan mengerjakan *UiPath developer advanced*. Ketika semua materi telah selesai maka sertifikat akan didapatkan sebagai tanda telah menyelesaikan semua topik yang ada pada *UiPath developer foundation*. Selanjutnya peserta dapat langsung mengerjakan *UiPath developer advance*.

### 3.2.2. Minggu 3 & 4 : Training RPA Developer Advanced

Pada minggu kedua magang dilakukan pelatihan *RPA developer advanced*. Pada pelatihan kali ini lebih terfokus untuk mempelajari *Robotic Enterprise Framework* (REFramework). REFramework merupakan sebuah kerangka kerja yang disediakan oleh UiPath yang dapat digunakan untuk membuat, mengetes, dan menjalankan proses bisnis. Gambar 3.3 tampilan REFramework memperlihatkan template dari REFramework pada UiPath.

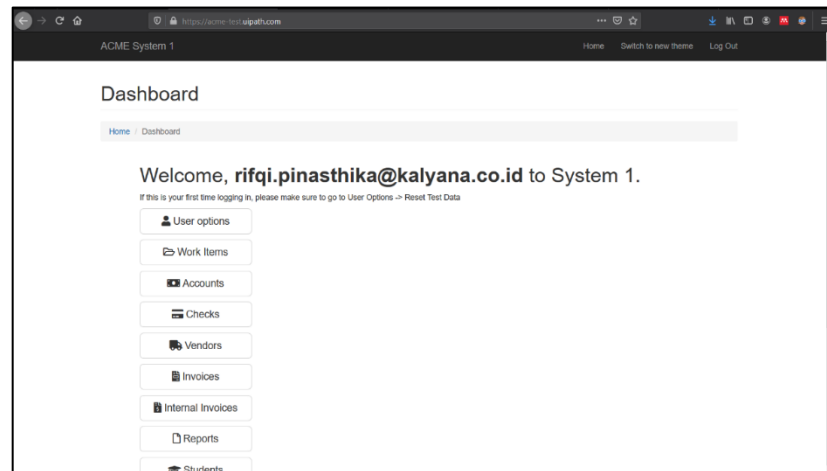


**Gambar 3. 3 Tampilan REFramework**

Untuk format pelatihan kedua ini sedikit berbeda dari pelatihan RPA *developer foundation*. Ada beberapa tahap yang harus dikerjakan oleh peserta untuk mengerjakan pelatihan ini. Pertama, peserta akan mempelajari terlebih dahulu mengenai REFramework melalui modul dan video yang ada. Kedua, peserta akan diberikan *process definition document* (PDD) yang berisi proses bisnis dan *workflow* dari kasus yang ada. Ketiga, peserta dapat membaca *hint* yang diberikan. Keempat, peserta dapat langsung membuat *workflow* dan submit hasilnya. Pihak UiPath akan menilai hasil dari *workflow* yang telah dibuat apakah berjalan sesuai atau tidak.

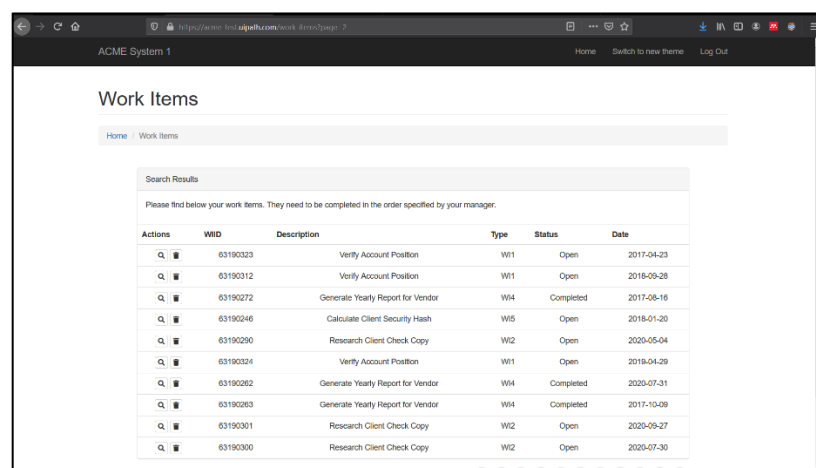
Untuk dapat menyelesaikan RPA *developer advanced* peserta wajib menyelesaikan dua permasalahan yang disediakan. Kasus pertama adalah *calculate client security hash*. Pada kasus ini peserta diharuskan untuk memperbarui detail dari item yang ada pada *website* [acme-test.uipath.com](https://acme-test.uipath.com). Untuk dapat mengakses *menu* yang harus diakses peserta diharuskan

membuat akun terlebih dahulu. Setelah memiliki akun maka peserta dapat *login* dan mengakses halaman utama yang diperlihatkan seperti gambar 3.4 *homepage website acme*.



**Gambar 3. 4 Homepage Website Acme**

Peserta akan mengakses *menu work items* yang tampilannya seperti pada gambar 3.5 tampilan *menu work items*. Setelah itu dapat dilakukan pengambilan data ClientID dan ClientCountry.



**Gambar 3. 5 Tampilan Menu Work Items**



Data yang telah didapat akan dienkripsi menggunakan *website* [sha1-online.com](http://sha1-online.com). Tampilan halaman *website* tersebut diperlihatkan seperti gambar 3.6 tampilan *website* sha1-online.



**Gambar 3. 6 Tampilan Website Sha1-online**

Data enkripsi akan digunakan untuk memperbarui detail dari *work items* dengan tipe WI5.

Kasus kedua yang dikerjakan berjudul *generate yearly report*. Peserta diminta untuk mengunggah laporan tahunan dari *work items* dengan tipe WI4. Untuk dapat mengunduh laporan maka dibutuhkan TaxID yang didapat dari detail *work items* bertipe 4. Laporan yang dapat diunduh adalah laporan bulanan melalui *menu download monthly report* seperti diperlihatkan oleh gambar 3.7 tampilan *menu download report*.

The screenshot shows a web browser window with the URL <https://acme-test.upath.com/reports/download>. The page title is "Reports - Download Monthly Report". Below the title is a breadcrumb trail: "Home / Reports - Download Monthly Report". The main content area contains a form with the following elements:

- A heading: "Please fill in the following form to be able to download a specific Vendor Monthly Report."
- A label "Vendor TaxID:" followed by a text input field containing "DE987564".
- A label "Month:" followed by a dropdown menu showing "January".
- A label "Year:" followed by a dropdown menu showing "2020".
- A blue button labeled "Download Report".

At the bottom of the page, there is a copyright notice: "Copyright © 2021 ACME Systems".

**Gambar 3. 7 Tampilan Menu *Download Report***

Laporan tahunan didapatkan dengan cara menyatukan laporan dari setiap bulan. Setelah laporan tahunan terbentuk maka dapat langsung diunggah melalui *menu upload report* dengan tampilan seperti gambar 3.8 tampilan *menu upload yearly report*.

The screenshot shows a web browser window with the URL <https://acme-test.upath.com/reports/upload>. The page title is "Reports - Upload Yearly Report". Below the title is a breadcrumb trail: "Home / Reports - Upload Yearly Report". The main content area contains a form with the following elements:

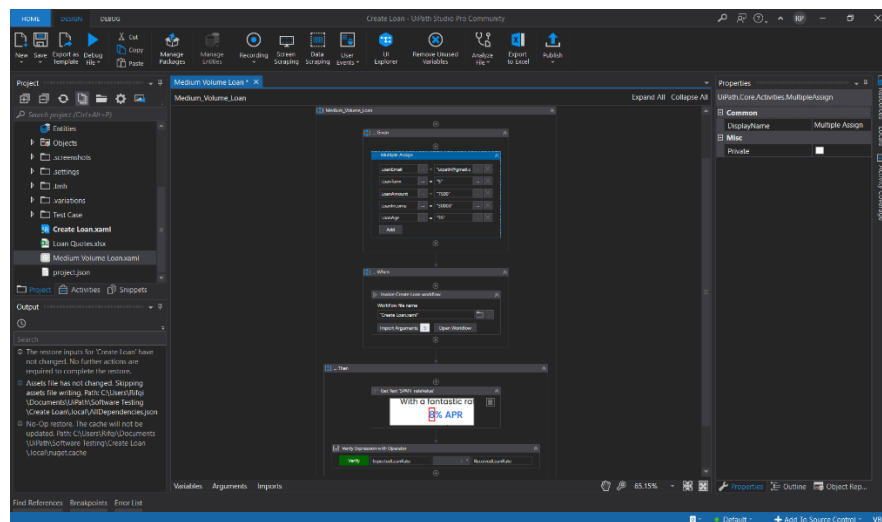
- A heading: "Please fill in the following form to be able to upload a specific Vendor Yearly Report."
- A label "Vendor TaxID:" followed by a text input field containing "DE987564".
- A label "Year:" followed by a dropdown menu showing "2020".
- A button labeled "Select Report File".
- A blue button labeled "Upload Report".

At the bottom of the page, there is a copyright notice: "Copyright © 2021 ACME Systems".

**Gambar 3. 8 Tampilan Menu *Upload Yearly Report***

### 3.2.3. Minggu 5 : Training UiPath Testing

Pada minggu kelima, dilakukan pelatihan untuk penggunaan fitur *testing* pada UiPath. *Testing* pada UiPath membutuhkan UiPath dengan *version pro*. *Testing* dapat dilakukan dengan cara membuat *test case* dari *workflow* yang telah dibuat dengan cara klik kanan pada *workflow* dan pilih *create test case*. Maka *test case* akan dibuat seperti terlihat pada gambar 3.9 tampilan *workflow testing*.



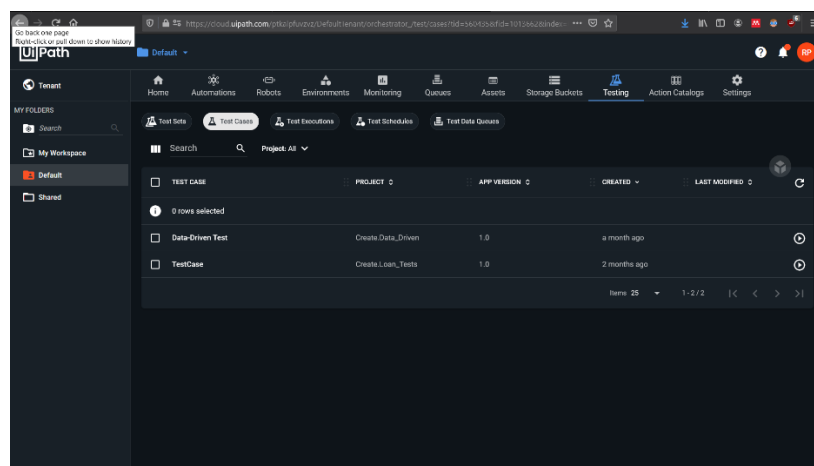
**Gambar 3. 9 Tampilan Workflow Testing**

*Test Case* UiPath akan secara otomatis membuat sebuah template dengan 3 bagian yaitu *given*, *when*, dan *then*. *Given* adalah tempat untuk mendeklarasikan dan menyiapkan data yang akan digunakan pada proses *testing*. Bagian *when* merupakan tempat *invoke workflow*. *Then* merupakan tempat untuk memverifikasi hasil output apakah sesuai atau tidak.

Selain pembuatan *test case* biasa UiPath juga mendukung *data driven testing*. *Testing* dengan jenis ini menggunakan sumber data untuk

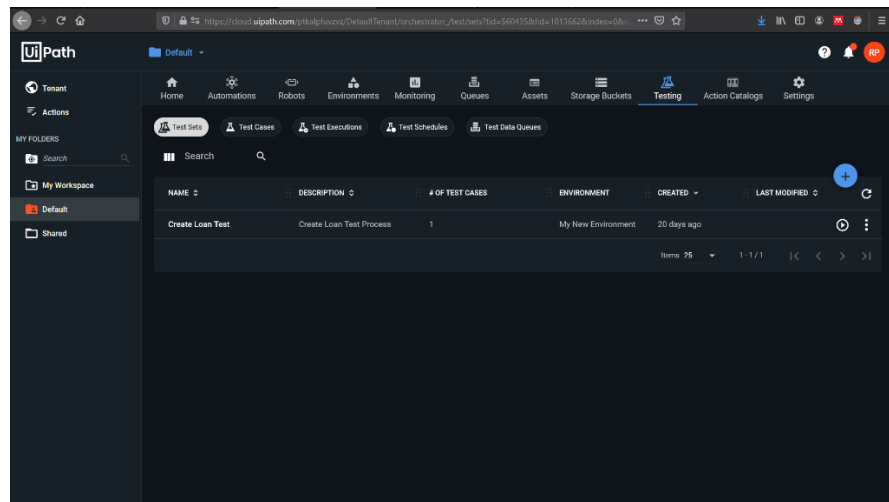
melakukan *testing*. Salah satu sumber data yang dapat digunakan adalah data dari Microsoft Excel. *Data driven testing* memungkinkan *testing* menggunakan banyak data dalam sekali eksekusi.

*Workflow testing* yang telah dibuat juga dapat dipublikasikan pada orchestrator. Orchestrator juga menyediakan fitur untuk menjalankan *testing*. Ada beberapa langkah untuk menjalankan *testing* melalui orchestrator. Pertama, *test case* pada UiPath Studio harus diunggah terlebih dahulu melalui UiPath Studio. Setelah diunggah *test case* yang telah dibuat akan muncul pada *menu testing* seperti terlihat pada gambar 3.10 tampilan *test case* Orchestrator.



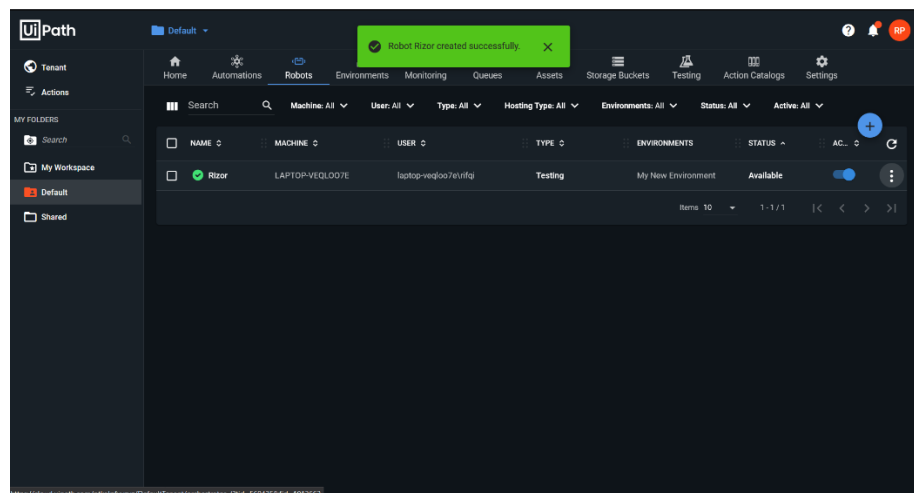
**Gambar 3. 10 Tampilan *Test Case* Orchestrator**

Kedua, *test set* harus dibuat terlebih dahulu di orchestrator menggunakan *test case* yang telah diunggah. Sebuah *test set* dapat memiliki beberapa *test case*. *Test set* yang telah dibuat akan terlihat seperti gambar 3.11 tampilan *test sets* Orchestrator.



**Gambar 3. 11 Tampilan *Test Sets* Orchestrator**

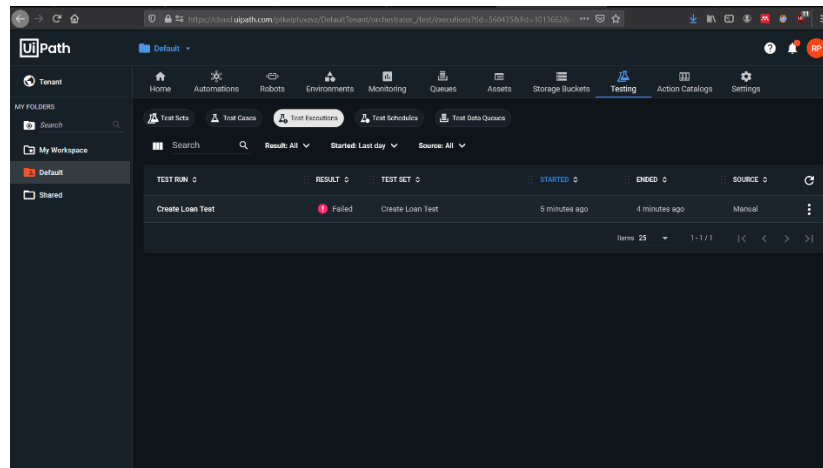
Terakhir, pastikan ubah tipe robot menjadi *testing*. Robot yang telah bertipe *testing* diperlihatkan seperti pada gambar 3.12 tampilan robot bertipe *testing*.



**Gambar 3. 12 Tampilan Robot Bertipe *Testing***

Setelah langkah-langkah sebelumnya telah dilakukan maka *test sets* dapat dijalankan. Setelah selesai dijalankan hasil dari *testing* tersebut akan

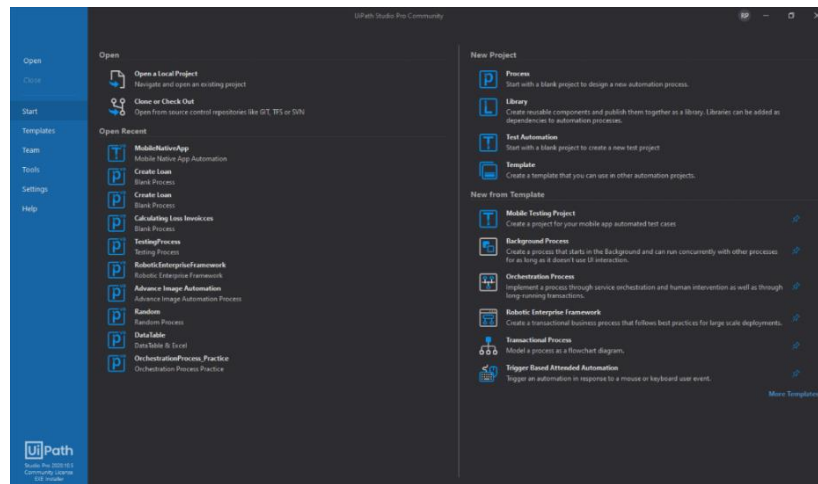
terlihat pada *test executions*. Tampilan hasil dari *testing* terlihat seperti gambar 3.13 tampilan *test executions* Orchestrator.



**Gambar 3. 13 Tampilan *Test Executions* Orchestrator**

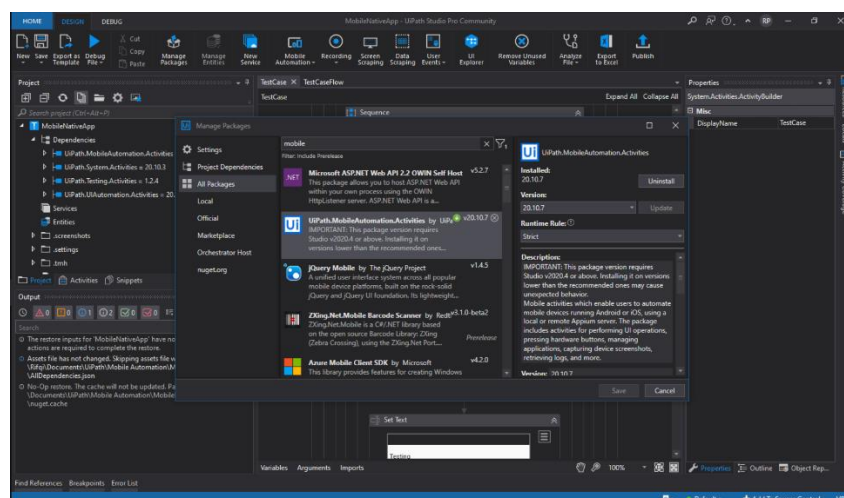
#### **3.2.4. Minggu 6 & 7 : Training UiPath Mobile Automation**

UiPath baru saja merilis fitur untuk automasi pada platform *mobile*. Fitur ini disebut UiPath *mobile automation*. Fitur ini mendukung automasi pada *mobile* dengan platform android dan iOS. Automasi *mobile* pada UiPath dapat dibuat dengan memilih *mobile testing* project pada halaman awal UiPath seperti pada 3.14 halaman awal UiPath Studio.



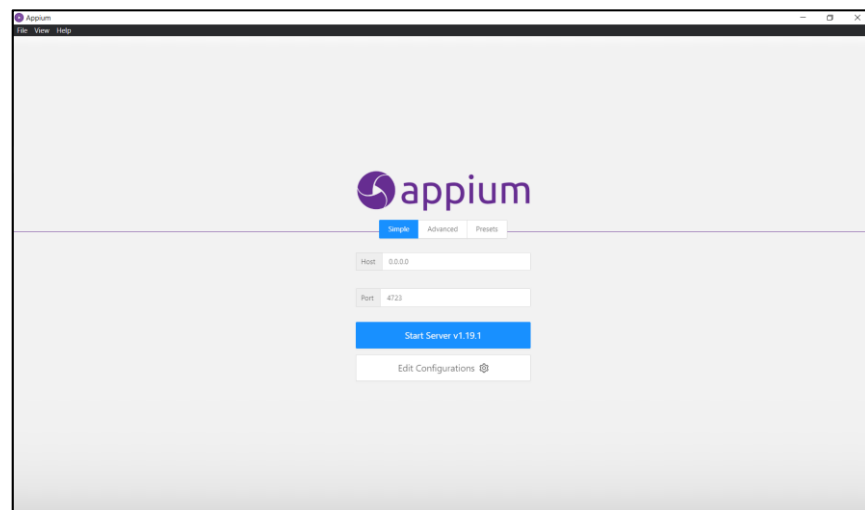
**Gambar 3. 14 Halaman Awal UiPath Studio**

Untuk menggunakan *mobile automation* dibutuhkan juga *dependencies* UiPath.Mobile.Automation.Activities. Untuk menambahkan *dependencies* tersebut maka diharuskan melakukan instalasi pada *manage packages*. *Dependencies* yang dibutuhkan dapat langsung dipasang pada *menu manage packages* seperti terlihat pada gambar 3.15 tampilan *manage package*.



**Gambar 3. 15 Tampilan *Manage Package***

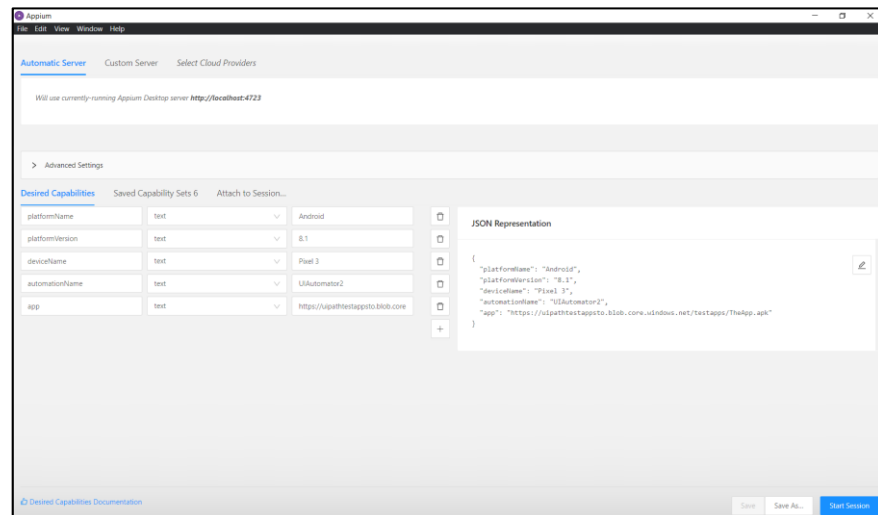
Setelah persiapan pada UiPath Studio telah dilengkapi maka dibutuhkan *device* untuk tes. *Device* yang digunakan adalah android dengan menggunakan emulator. Untuk menggunakan *device* android dibutuhkan java, jdk, jre, dan appium. Setelah kebutuhan dilengkapi maka dapat langsung membuka *software* appium yang tampilannya seperti gambar 3.16 tampilan Appium.



**Gambar 3. 16 Tampilan Appium**

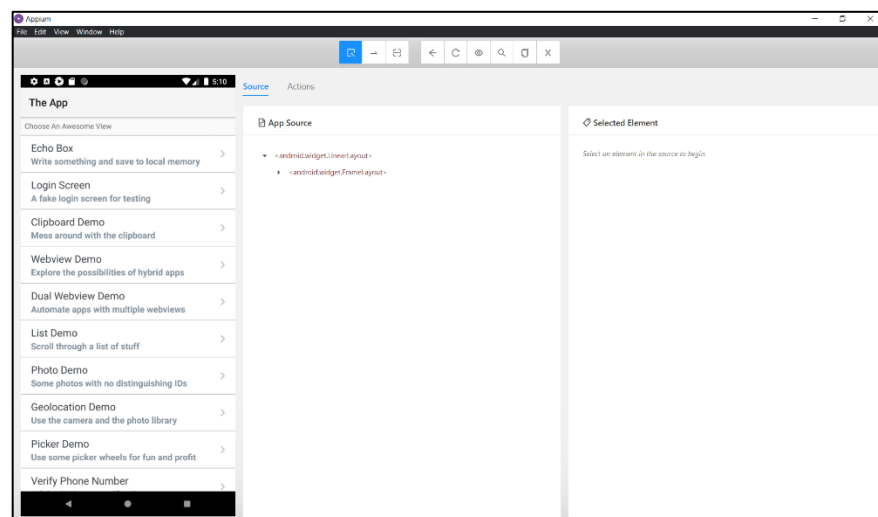
Appium merupakan *framework* yang dapat melakukan automasi pada platform *mobile native*, *hybrid* dan aplikasi web pada android, iOS dan firefoxOS (Momin, 2013). Untuk menghubungkan appium dengan *device* ada beberapa data yang dibutuhkan seperti *platformname*, *platformversion*, *devicename*, dan lainnya seperti terlihat pada gambar 3.17 tampilan *setup menu* Appium.





**Gambar 3. 17 Tampilan Setup Menu Appium**

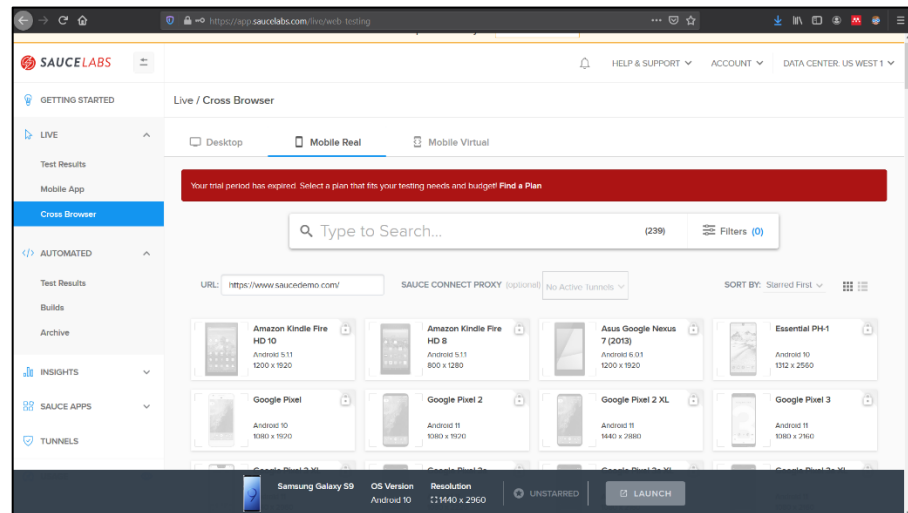
Untuk mengetes koneksi antara appium dengan *device* dapat diketahui dengan *start session*. Ketika koneksi berhasil maka akan muncul tampilan dari *device* yang digunakan seperti ditampilkan pada gambar 3.18 tampilan berhasil terhubung Appium.



**Gambar 3. 18 Tampilan Berhasil Terhubung Appium**

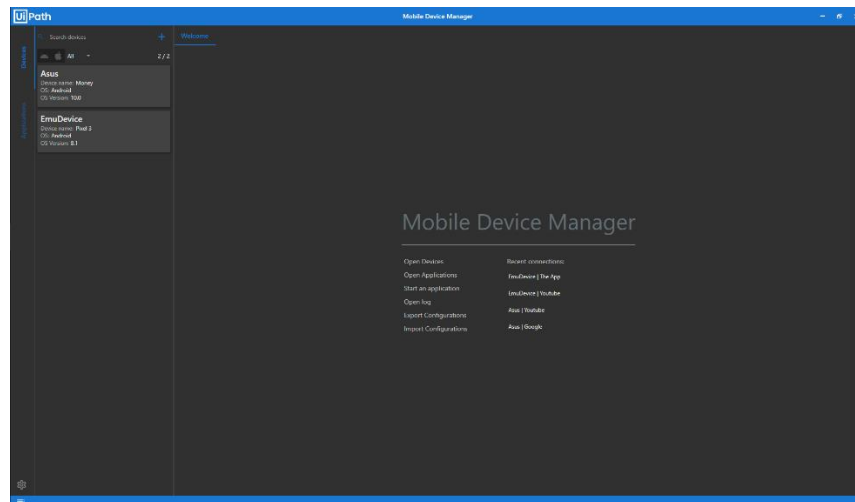
*Device* dengan menggunakan *cloud emulator* juga dapat digunakan selain penggunaan *device local* dengan menggunakan appium. Salah satu

penyedia *cloud emulator* adalah saucelabs. Saucelabs menyediakan berbagai macam *device* untuk digunakan sebagai media *testing*. Tampilan dari saucelabs terlihat pada gambar 3.19 tampilan Saucelabs.



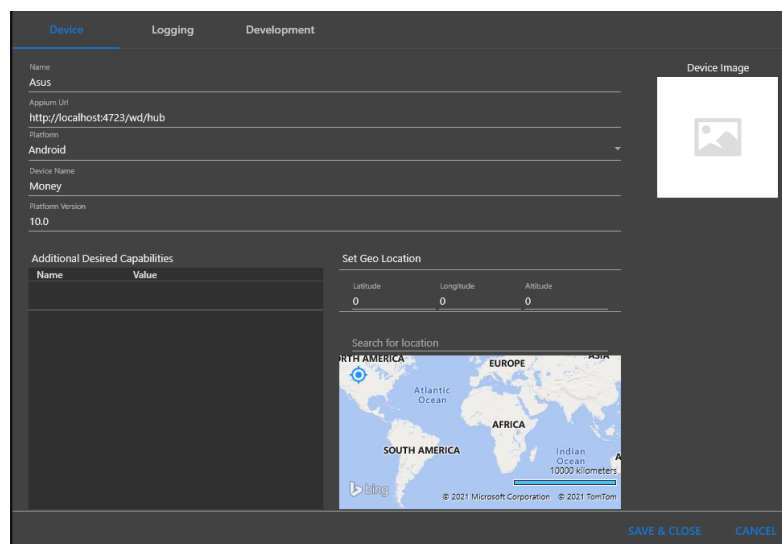
**Gambar 3. 19 Tampilan Saucelabs**

Setelah semua persiapan lengkap maka UiPath *mobile automation* dapat digunakan. Untuk mengatur *device* dan aplikasi yang digunakan dapat melalui *menu mobile device manager* yang tampilannya seperti gambar 3.20 tampilan *mobile device manager*.



**Gambar 3. 20 Tampilan *Mobile Device Manager***

Gambar 3.21 tampilan *setup device* memperlihatkan *menu* untuk menambahkan *device*. Beberapa data yang dibutuhkan adalah nama, url dari appium, platform, nama *device*, dan versi.



**Gambar 3. 21 Tampilan *Setup Device***

Untuk aplikasi ada dua pilihan aktivitas yang akan dijalankan yaitu melalui aplikasi atau web. Aplikasi memerlukan data nama dari *package* aplikasi dan nama dari *activity*. Untuk web membutuhkan nama dari

browser dan *link* dari *website* yang ingin diakses. Tampilan dari pengaturan aplikasi dapat dilihat pada gambar 3.22 tampilan *setup application*.

**Edit application**

Name  
Youtube

☒ App ☐ Web

Android iOS

App

App package  
com.google.android.youtube

App activity  
.app.honeycomb.Shell\$HomeActivity

Start Page

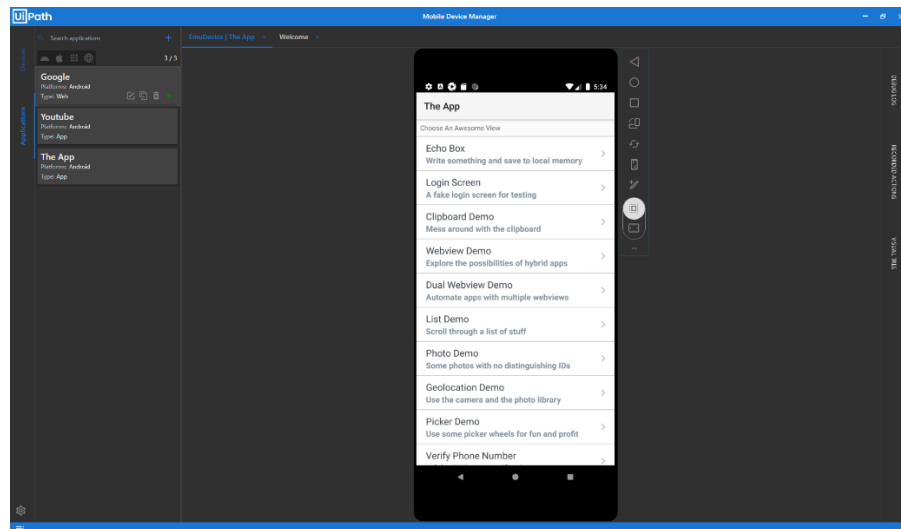
Additional Desired Capabilities

Name	Value

SAVE & CLOSE CANCEL

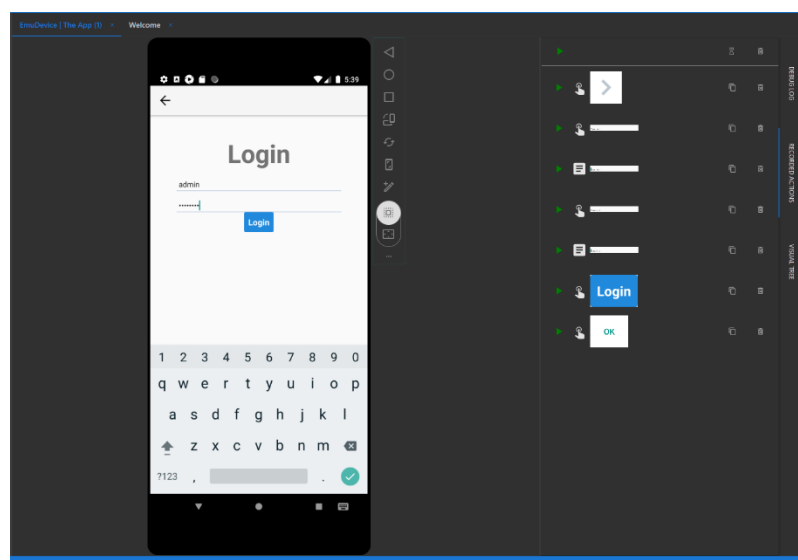
**Gambar 3. 22 Tampilan *Setup Application***

Setelah selesai mengkonfigurasi *device* dan aplikasi maka aplikasi dapat dijalankan. Saat aplikasi dijalankan akan muncul tampilan dari *device* yang digunakan pada *mobile device manager* seperti pada gambar 3.23 tampilan *device* terhubung dengan UiPath.



**Gambar 3. 23 Tampilan *Device* Terhubung Dengan UiPath**

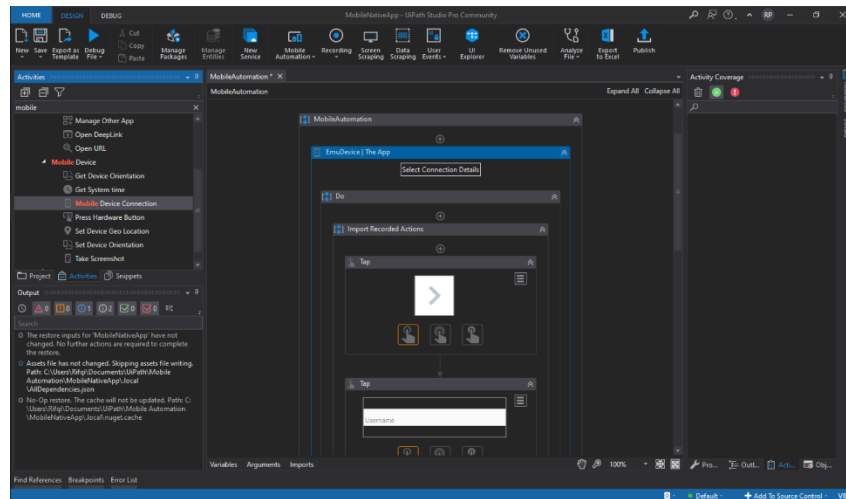
Semua aktivitas yang dilakukan pada *device* akan terekam. Rekaman aktivitas tersebut berada pada *tab recorded actions* yang terlihat seperti gambar 3.24 *record mobile activity*.



**Gambar 3. 24 *Record Mobile Activity***

Rekaman aktivitas yang telah dilakukan sebelumnya dapat dimasukkan kedalam *workflow*. Gambar 3.25 *import mobile activity*

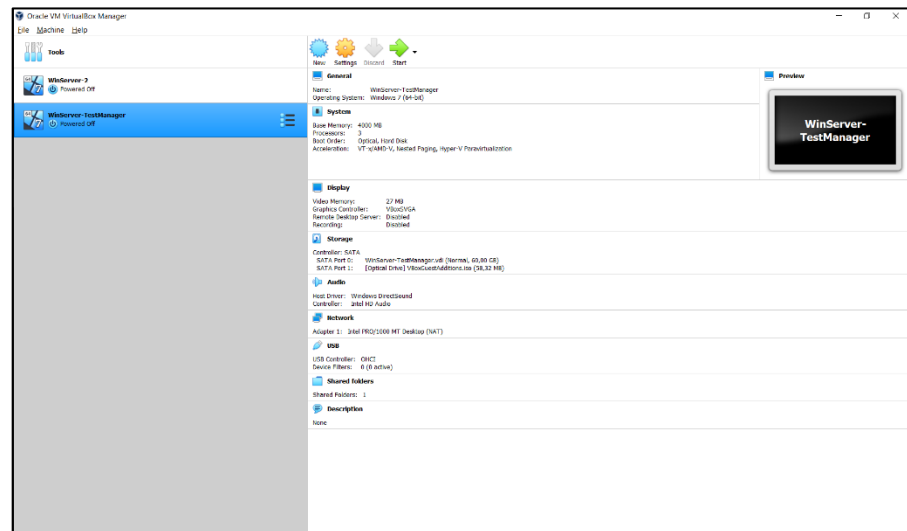
memperlihatkan tampilan dari aktivitas yang telah direkam yang telah menjadi *workflow*.



**Gambar 3. 25 Import Mobile Activity**

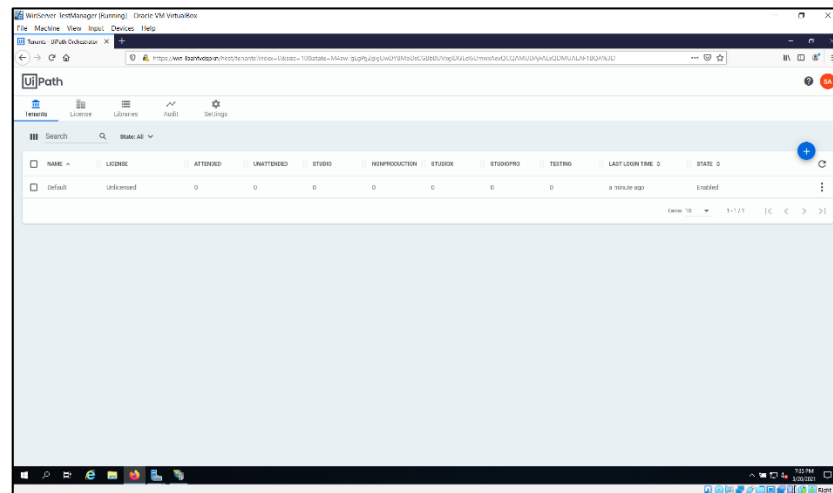
### 3.2.5. Minggu 8 : Mempersiapkan Orchestrator On Premise

Pada minggu ke delapan dilakukan instalasi untuk orchestrator *on premise*. Instalasi ini merupakan persyaratan untuk menggunakan UiPath Test Manager yang selanjutnya akan digunakan dalam proyek. Untuk instalasi ini ada beberapa persyaratan yang meliputi syarat *hardware* dan *software* yang tercantum pada *website* UiPath.



**Gambar 3. 26 Home Virtual Box**

Untuk instalasi ini digunakan *virtual machine* dengan *software* virtual box yang tampilannya seperti gambar 3.26 *home Virtual Box*. Penggunaan virtual machine dikarenakan harus ada instalasi *operating system* Windows Server dengan versi minimal 2012 untuk menjalankan orchestrator *on premise*. Saat Windows Server sudah terpasang maka dapat dilakukan instalasi fitur tambahan. Selain itu juga dibutuhkan Microsoft Sql Server sebagai basis datanya. Setelah semua persyaratan terpenuhi maka instalasi dapat dilakukan. Berikut adalah tampilan dari orchestrator *on premise* diperlihatkan oleh gambar 3.27 tampilan Orchestrator *on-premise*.

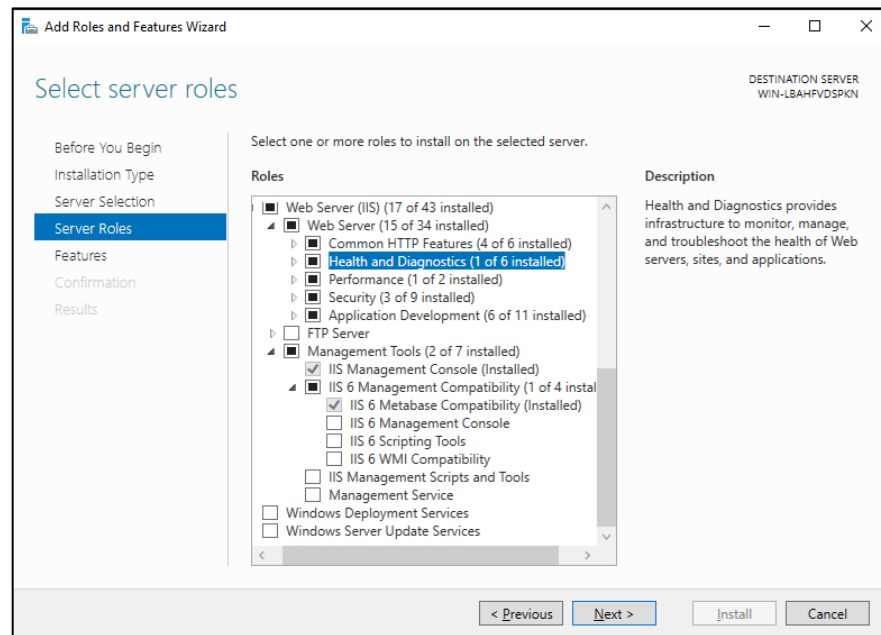


**Gambar 3. 27 Tampilan Orchestrator *On-Premise***

### **3.2.6. Minggu 9 & 10 : Mempersiapkan UiPath Test Manager**

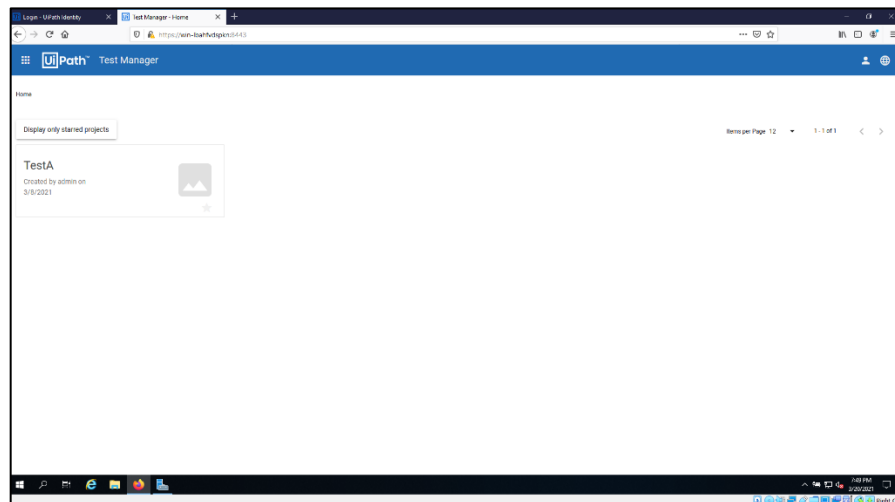
Pada minggu sebelumnya salah satu persyaratan instalasi UiPath Test Manager telah diselesaikan. Minggu ini dapat dilanjutkan dengan melengkapi persyaratan lainnya untuk penggunaan aplikasi ini. Hal pertama adalah dengan menambahkan beberapa fitur pada Windows Server pada *menu* seperti terlihat pada gambar 3.28 Windows Server *add features*.





**Gambar 3. 28 Windows Server *Add Features***

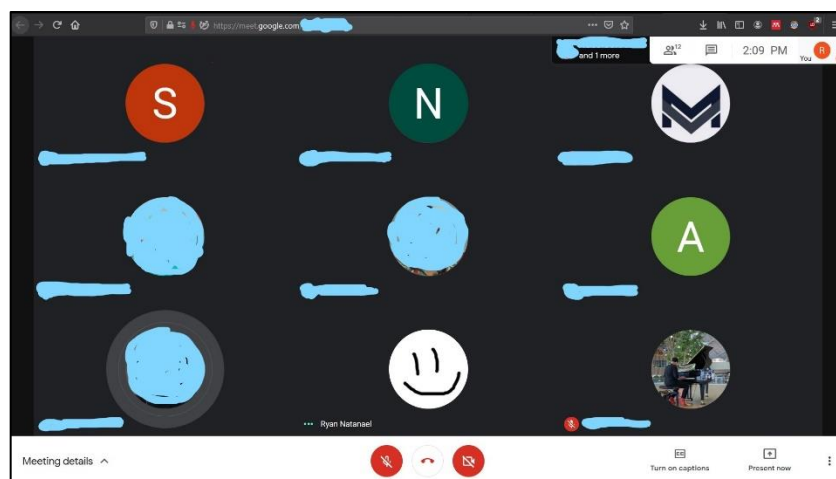
Setelah semua fitur lengkap maka tinggal beberapa persyaratan yaitu konfigurasi pada Windows. Konfigurasi tersebut meliputi mematikan autentikasi Windows untuk Test manager dan pembuatan *SSL certificate*. Setelah instalasi akan muncul tampilan UiPath Test Manager pada browser seperti terlihat pada gambar 3.29 tampilan UiPath Test Manager.



**Gambar 3. 29 Tampilan UiPath Test Manager**

### **3.2.7. Minggu 11 : *Meeting POC Dengan Client***

Pada minggu ini dilakukan pertemuan dengan client dari PT XYZ. Meeting dilakukan dengan menggunakan platform dari Google yaitu Google Meet seperti terlihat pada gambar 3.30 *meeting POC via Google Meet*.



**Gambar 3. 30 Meeting POC Via Google Meet**

Pada pertemuan ini dilakukan pembahasan mengenai pemberian akses jaringan *client* dan *remote desktop* agar dapat diakses oleh pihak Kalyana.

Akses ke komputer *client* berguna untuk melakukan persiapan aplikasi yang akan digunakan dalam proyek yang akan berjalan.

#### **3.2.8. Minggu 12 : Persiapan Proyek Dengan Divisi IT *Client***

Sebelum dapat melakukan instalasi program yang akan digunakan pihak Kalyana masih *menunggu* persiapan dari pihak IT *client*. Setelah persiapan *client* selesai pihak Kalyana diberikan kode akses untuk mengakses jaringan privat yang digunakan oleh pihak *client*. Setelah terhubung dengan jaringan privat *client* instalasi dapat dilakukan untuk mempersiapkan demo proyek.

### **3.3. Kendala Yang Dihadapi**

Beberapa kendala yang dialami selama melakukan kerja magang di PT Kalyana Adikarya Abadi adalah sebagai berikut :

- a. Kendala saat menyelesaikan sertifikasi dikarenakan baru pertama kali menggunakan UiPath.
- b. Penggunaan bahasa pemrograman yaitu VB .Net yang digunakan dalam UiPath.
- c. Kendala saat instalasi *software* UiPath Test Manager dikarenakan merupakan fitur yang baru saja rilis.

### **3.4. Solusi Atas Kendala**

Berikut dijelaskan bagaimana agar dapat menyelesaikan kendala yang dialami selama kerja magang :

- a. Bertanya kepada pembimbing maupun rekan kerja serta mencari dari sumber lain seperti youtube.
- b. Bertanya kepada pembimbing serta membuka kembali dokumentasi VB .NET.
- c. Bertanya di forum resmi UiPath yang kemungkinan akan dijawab oleh pihak UiPath.